

面向重用的构件化平台设计和实现

杨 晔

(浙江警官职业学院, 浙江 杭州 310018)

摘 要: 提出了利用抽象工厂模式来解决构件化平台的设计架构, 并给出了构件化平台的描述和讲解了构件组装, 给出了 CACP 的实现细节和实现技术. 通过实践证明, CACP 确实可以对技术积累和产品化路线的规划起到推动作用, 并且大幅提高实际开发中的开发效率.

关键词: 重用; 公共应用构建平台; 构件; 抽象工厂模式

中图分类号: TP31

文献标识码: A

文章编号: 1000-7180(2010)05-0209-03

Reuse-oriented Component-Based Platform Design and Implementation

YANG Ye

(Zhejiang Police Vocational Academy, Hangzhou 310018, China)

Abstract: This paper describes the basic theory of components and proposed to use an abstract factory pattern to solve the architecture design of the component-based platform. It also gives a description of component-based platform, and explained the assembling of components, and gives the implementation details and technologies of CACP. It is proved by practice that CACP could promote the technology accumulation and the product lines, and improve the efficiency of the actual software development greatly.

Key words: reuse; CACP; component; abstract factory

1 引言

目前, 采用软构件思想开发的软件产品相继出台, 其技术应用逐渐成熟, 各种开发框架、平台代表了这个行业的发展方向. 在国内从事企业级应用软件的开发商有了最基本的基础业务框架(包含平台框架、组织、权限), 但是整体上其解决方案比较弱, 应用效果不够好, 其通用性和稳定性还需市场的检验, 还需要逐渐地把基础业务框架(包含平台框架、组织、权限)、工作流系统、报表工具、商业智能集成在一起, 形成一个产品, 为业务系统的开发提供一套整体的解决方案.

构件^[1](Component)是可复用的软件组成成分, 可被用来构造其他软件. 它可以是被封装的对象类、类树, 一些功能模块、软件框架、软件构架(或体

系结构)、文档、分析件、设计模式等. 从广义上讲, 软构件技术是基于面向对象的, 以嵌入后马上可以使用的即插即用型软构件概念为中心, 通过构件的组合来建立应用的技术体系. 狭义上讲, 它是通过构件组合支持应用的开发环境和系统的总称.

根据构件的基本属性, 如何设计一个构件系统, 首先要确定构件技术的关键、构件交互的规则、构件存在的环境, 采用支撑性的技术辅助我们完成一个构件化的平台.

2 基于抽象工厂模式的构件技术

抽象工厂模式的物理模型^[2]如图 1 所示, 主要包括以下几个部分:

(1) 抽象工厂角色 AbstractFactory. 它是工厂方法模式的核心, 负责定义创建抽象产品对象的接

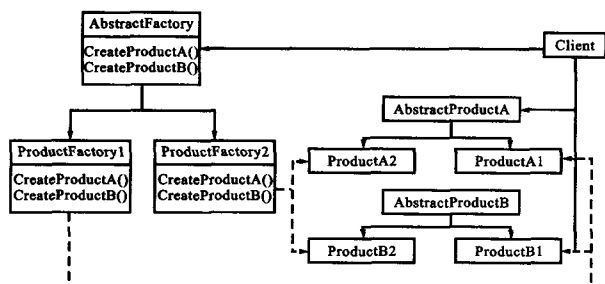


图1 抽象工厂模式的物理模型

口,任何在模式中用于创建具体实体产品的工厂类都必须从抽象工厂派生,并且实现由它定义的所有工厂方法。

(2) 抽象产品角色 AbstractProductA, AbstractProductB. 它负责提供实体产品 ProductA 和 ProductB 的公共接口。

(3) 实体工厂角色 ProductFactory1 和 ProductFactory2. 它是抽象工厂模式的对外接口,它负责创建对象的内部逻辑,直接在客户端的调用下创建产品的实例,可以被外界直接调用。

(4) 实体产品角色 ProductA1, ProductA2, ProductB1, Product B1. 它的抽象工厂模式最终的创建目标,对应着某一个具体产品类的实例,在这里我们把它对应成为一个个构件。

但 AbstractFactory 模式自身扩展有限,难以生产新种类产品,因此需要改造该模式,以实现构件化平台的需求。

该模式在构件化平台里面是使用三层架构来实现,在服务层和数据层分别定义抽象工厂和抽象产品,定义构件插拔的接口。然后定义不同的各层构件,如图2所示。

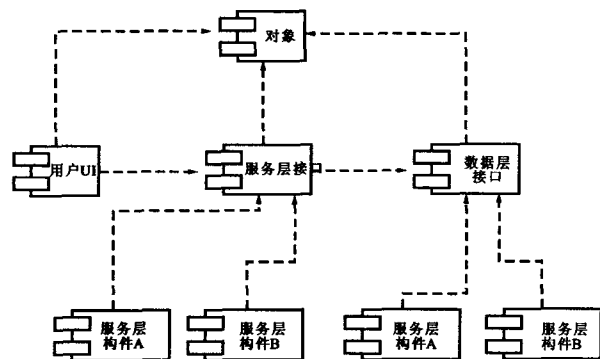


图2 构件模型图

通过这种抽象工厂,使构件对外发生作用或构件间的交互,都是通过规范定义的接口进行,构件使用者只需要知道这些接口,而不关心其内部是如何

实现的,这是设计与实现分开的关键.因此这个架构就应当提供构件交互的规则,并基于这些规则实现类似容器的标准环境.即实现一个类似于电脑主板的平台,提供一个个插槽供各种设备插入.同时,构件内的每个服务都是唯一的,而且不会和其他构件内的服务重复,其实就是要遵循 OO 提出的单一职责和开放——封闭准则,即实现一个类似于电脑主板的平台,提供一个个插槽供各种设备插入.同时,每个插在板子上的设备都提供了唯一的功能,最后组成了一台可以使用的电脑。

3 平台实现

本平台分为三个层次^[3],最底层是框架层,主要包括权限、 workflow、组织架构;次底层为基础应用层,主要包括自定义开发模块、可配置生成的报表统计、查询统计等应用的通用模块;上层为应用层。

结合前期 IBatisNET vsADO.NET^[4]性能测试和 WcfVsRemoting^[5]传输测试记录这些技术预研成果,考虑到平台的稳定性、兼容性、项目风险管理、企业信息化桌面应用软件的普及程度等因素,平台采用.NET 2.0 的 WinForm(桌面应用开发)和 WebForm(ASP.NET)开发技术实现,其中 WinForm 表示层和业务逻辑层之间的通讯采用 Remoting 远程对象调用技术实现而不是本地对象的引用,WebForm 对服务层逻辑的调用采用本地对象调用方式。

结合分层架构,给出平台技术架构框图,如图3所示。

需要详细说明的有以下几点:

(1) 用 cookie 存储身份认证信息(比如登录用户 ID).更复杂的用户信息描述采用 xml 格式字符串来描述,比如,用户 ID、姓名,所在单位 ID、姓名。

(2) ASP.NET WebForm 直接调用服务层业务逻辑组件,但 Form 与服务层逻辑的交互要求传递可序列化对象,以便后期 WinForm + Remoting 方式开发能够对业务逻辑进行的复用和便于搭建统一服务逻辑组件中心的搭建,方便客户端(WebForm 和 WinForm)进行服务逻辑统一调用。

(3) ASP.NET WebForm 采用 CodeBehind 方式进行页面开发.主要是方便页面间模块化和界面逻辑间的解耦,并利用 ASP.NET 本身特性方便实现 web 页面构件化(热插拔)。

(4) 远程对象服务中心是个构件化的容器,将服务组件以构件的方式进行远程对象的部署,主要给 Winform 客户端软件使用。

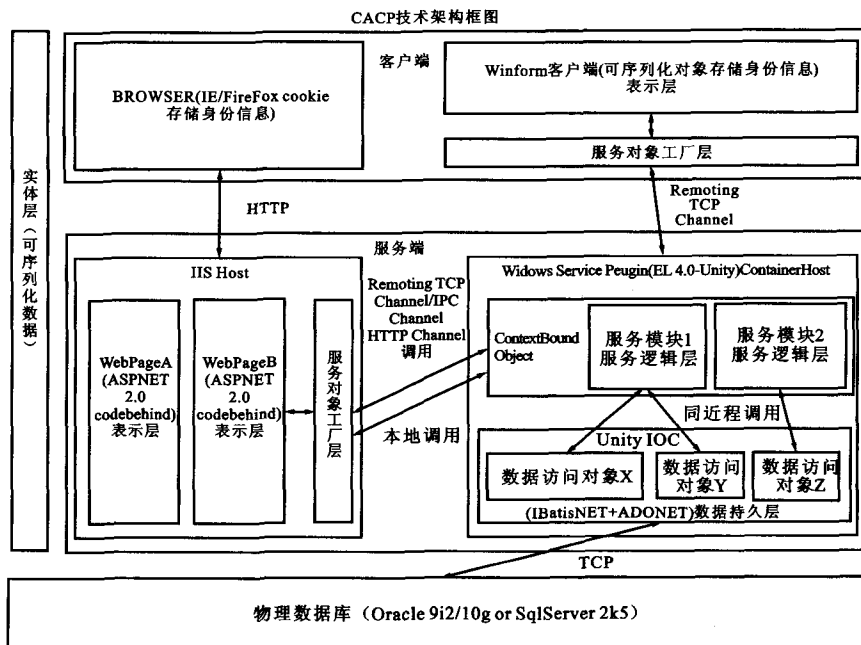


图 3 平台技术架构图

(5) 为了避免悲观锁带来的死锁导致维护不便的问题,本系统并发锁控制方式采用 Hibernate^[6] 推荐的乐观锁控制方式,就是每个表都加个 versionId (VId)版本号,每次修改的时候更新这个版本号,修改前判断某记录是否版本号变化过,变化过则说明被其他人修改过,程序进行提示或者其他处理.需要加入并发锁的地方都要适当加入并发锁控制程序逻辑.

(6) 服务类都要继承 System. ContextBound-

Object 对象,主要是为了 RealProxy 方式实现面向方面的编程(AOP)^[7],可以将业务无关的异常处理代码、日志、数据库事务处理等代码进行统一处理机制,可以减少大量重复代码.

(7) 总体架构设计

平台按功能层次总体分为三层,如图 4 所示.其中加粗字体部分为项目范围内功能模块,其他为扩展示意.

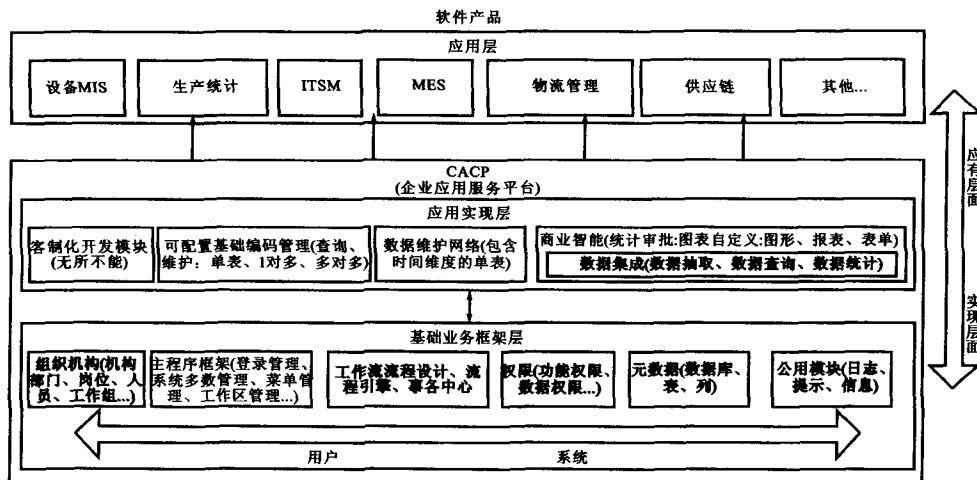


图 4 总体架构设计图

基础业务框架层包含的功能是项目计划的第一阶段内容,也是所有应用平台实现的基石;应用实现层实在基础框架层之上扩展的一些通用模块或定制化模块.数据集成、统计报表、统计查询等商业智

能这部分功能在项目第二阶段进行实现;中间层非加粗字体部分和最上面的应用层则为利用平台实现的应用系统所在层次,属于平台范围之外.

(下转第 216 页)

泛的关注和应用,文中考虑了词条的时序特征,采用近期出现的垃圾词条辅助垃圾邮件判别过程,实验结果表明,文中方法较传统贝叶斯垃圾邮件过滤更加有效.基于机器学习的邮件过滤技术派生出了多个分支,使用决策树等其他经典机器学习技术处理垃圾邮件过滤,将是下一步研究的方向.

参考文献:

- [1] Androutsopoulos I, Koutsias J, Chandrinou G, et al. An evaluation of naive bayesian anti-spam filtering[C]// Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning. Barcelona, Spain, 2000:9-17.
- [2] Androutsopoulos I, Koutsias J, Chandrinou K, et al. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with encrypted personal email messages[C]// Procs of 23rd International ACM SIGIR Conference on Research and Development in Information Retrieval. Greece, Athens, 2000:160-167.
- [3] Sakkis G, Androutsopoulos I, Paliouras G, et al. A memory-based approach to anti-spam filtering for mailing lists[J]. Information Retrieval, 2003,6(1):49-73.
- [4] 李翔鹰,叶枫.一种基于多贝叶斯算法的垃圾邮件过滤方法[J].计算机工程与应用,2006(31):114-116.
- [5] 廖明涛,张德运,李金库.基于朴素贝叶斯和层次聚类的两阶段垃圾邮件过滤方法[J].微电子学与计算机,2007,24(8):1-4.
- [6] Sarah J D, Padraig C, Alexey T, et al. A case-based technique for tracking concept drift in spam filtering[C]// Applications and Innovations in Intelligent Systems XII. Procs. of AI, England:Cambridge, 2004:3-16.
- [7] Androutsopoulos I, Koutsias J, Chandrinou K V, et al. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with encrypted personal e-mail messages[C]// Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2000). Athens, Greece,2000:160-167.

作者简介:

常青女,(1980-),助理实验师.研究方向为网络信息安全、计算机应用.

(上接第211页)

工厂层在三层架构中不算实体层次,只是为了方便客户端调用而存在的一个服务对象外观类.

4 结束语

面向重用的构件化平台的开发,将从根本上提高软件开发的效率,被认为是一种有效的解决软件危机的手段,并接受新兴软件制造业带来的挑战,但是,目前该研究还处在开始阶段,很多想法还不成熟,如何用更好的设计思想来完善构件化平台,对企业级应用提供更好的技术支撑,是我们下一步研究的重点.

参考文献:

- [1] 杨芙清,梅宏.构件化软件设计与实现[M].北京:清华大学出版社,2008.
- [2] Erich Gamma, Richard Helm, Ralph Johnson, et al. 设计模式——可复用面向对象软件的基础[M].北京:机械工业出版社,2007.
- [3] Rockford Lhotka. Expert C# 2005 business objects[M].北京:电子工业出版社,2006.
- [4] Clinton Begin, Brandon Gooding, Larry Meadors. Ibaitis in Action[M]. USA, Greenwich: Manning Publications, 2007.
- [5] Juval Lowy, Clemens Vasters. Programming WCF services[M]. USA, Sebastopol, O'Reilly Media,2008.
- [6] Robert E Filman, Tzilla Elrad, Siobhan Clarke, et al. Aspect-oriented software development[M]. Boston: Addison-Wesley,2005.
- [7] Christian Bauer, Gavin King. Hibernate 实战[M].北京:人民邮电出版社,2008.

作者简介:

杨晔女,(1978-),讲师.研究方向为信息管理技术、软件开发及测试技术.